UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR

GRADO EN INGENIERÍA INFORMÁTICA

TRABAJO FIN DE GRADO

PLATAFORMA DE CONCIENCIACIÓN FRENTE A ATAQUES DE INGENIERÍA SOCIAL MEDIANTE USB´S

ESTUDIANTE: Alejandro Moreno San Vidal

PROFESOR: José Luis García Dorado

PONENTE: Jorge López de Vergara

Madrid, Octubre / Noviembre 2018

AGRADECIMIENTOS

Me gustaría agradecer a todas las personas que me han ayudado y apoyado para hacer posible este proyecto:

A mi padre y mi madre, por su dedicación y esfuerzo para que yo haya podido llevar este proyecto adelante, sin ellos nada de esto habría sido posible. Gracias a vosotros estoy acabando mi carrera, y sé que es un orgullo para vosotros el saber que al final he conseguido sacarlo.

Me gustaría también agradecer el apoyo incondicional de mi hermano, que nunca ha fallado en momentos difíciles y espero que le pueda servir como ejemplo que a base de esfuerzo y sacrificio todo se consigue. Gracias hermano.

Agradecer la ayuda en todo momento de mi tutor José Luis García Dorado, que me ha ido guiando y ayudando para que pudiese sacar este trabajo adelante satisfactoriamente.

Dar gracias a todos los profesores que me han ayudado y aportado muchos conocimientos que me van a ser de gran ayuda durante toda mi vida.

Finalmente, me gustaría hacer mención a una persona muy importante en mi vida, que sé que le habría hecho mucha ilusión ver que he podido sacarlo. Siempre me dio apoyo y estuvo a mi lado en todo momento y sobre todo me dio una lección de que hay que luchar en esta vida hasta el final y no darse nunca por vencido. Fuiste único e irrepetible, y todo este esfuerzo va por ti. Muchas gracias abuelo.

RESUMEN

Resumen:

El presente trabajo de Fin de Grado consiste en la realización de una plataforma para llevar a cabo campañas de Phishing orientadas a empresas/colectivos. Dicha plataforma permite evaluar el nivel de concienciación que puede llegar a tener un ataque de ingeniería social, en este caso, mediante USB´s.

La ingeniería social consiste en la manipulación psicológica de las personas para que compartan información confidencial o hagan acciones inseguras. Este tipo de ataques se han convertido en una gran amenaza para todas las comunidades virtuales y es uno de los medios más efectivos para atacar a los sistemas de información, ya que atentan contra el factor más débil de seguridad: el factor humano. En este caso específico, el atacante utiliza etiquetas llamativas con palabras clave, de tal forma que se pone a prueba el nivel de codicia y curiosidad de las personas. Para evitar que se caiga en estas acciones inseguras se necesita realizar un entrenamiento o sistema de conciencia para poder defenderse contra ellos y así poder evitar cualquier tipo de riesgo que pueda comprometer a una institución o incluso a la propia persona de forma individual.

El presente documento muestra un sistema completo cuyo objetivo es ayudar a esta concienciación para que mejoren la toma de decisiones a través de la educación y las pruebas.

Concretamente, se ha hecho uso del lenguaje C# para la programación del archivo ejecutable, mientras que para la interfaz gráfica se ha utilizado el lenguaje HTML, Css y JavaScript con respecto al desarrollo front-end y del lenguaje SQL y Php para el desarrollo back-end.

Palabras clave:

Concienciación, ingeniería social, USB, nube, protocolo HTTP, sistemas de información, criptografía simétrica, interfaz gráfica, desarrollo front-end, desarrollo back-end.

ABSTRACT

This Bachelor Thesis consists in the realization of a platform for companies / groups in order to evaluate the level of awareness that a social engineering attack can have, in this case through USB's.

Social engineering is the psychological manipulation of people to share confidential information or make unsafe actions. This type of attack has become a great threat to all virtual communities and is one of the most effective means of attacking information systems, as they attack the weakest security factor: the human factor. In this specific case, the attacker uses striking tags with keywords, in such a way that the level of greed and curiosity of the people is put to test, so that they need to be trained to defend against these attacks and thus be able to avoid any type of risk that could compromise an institution or even them.

This document shows a tool in the cloud whose objective is to help raising awareness among people to improve decision-making through education and testing. For the realization of this study, it has been necessary to know the programming, configuration and hiding of executable files (.exe), as well as the elaboration of a graphical administration interface of the extracted data and the understanding of the HTTP protocol and the symmetric cryptography to be able to establish a secure connection with the server.

Specifically, the C # language has been used for the programming of the executable file, while for the graphical interface the HTML, Css and JavaScript language has been used with respect to the front-end development and the SQL and Php language for back-end development.

Keywords:

Awareness, social engineering, USB, cloud, HTTP protocol, information systems, symmetric cryptography, graphical interface, front-end development, back-end development.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS

1. INTRODUCCIÓN

En este último siglo nos hemos acostumbrado al uso de los medios de comunicación en nuestro día a día, lo hemos convertido en una costumbre e incluso, en una forma de vida. Uno de los medios que más utilizamos es el ordenador, con el que estamos constantemente realizando cambios de información. Este uso ha sido tan rápidamente extendido que en muchas ocasiones no tenemos conciencia de la forma en la que lo utilizamos, pudiendo llegar a tener un alto grado de sensibilidad. Por ello, debemos prestar mucha atención a todos los datos que manejamos, ya que algunos de estos pueden ser muy perjudiciales.

Las unidades de memoria USB o *pen drives* son unos de los dispositivos más usados popularmente para la transferencia de datos y uno de los datos de estudio a lo largo del proyecto. Se trata de una de las opciones más escogidas entre los ciudadanos por su comodidad y facilidad de manejo. Sin embargo, estos dispositivos también es la opción preferida para los ciberdelincuentes, utilizando estos dispositivos para su interés.

La investigación se realizó por el interés de conocer más detalladamente los peligros a los que nos podemos exponer, estudiando a su vez, las vulnerabilidades que aportan tanto los factores tecnológicos cómo los humanos.

El objetivo de dicho proyecto, por lo tanto, es dar cierto conocimiento para evitar ataques cibernéticos a través de estos pequeños dispositivos y así evitar graves problemas, no sólo a nivel individual, sino empresarial. Los ciberdelincuentes utilizan un patrón muy similar: dejan estos USB´s como “objetos perdidos” y a su vez, como medio de infección en ciertas zonas concretas; en el momento que un individuo encuentre uno de estos y lo conecte a su máquina, sin que se dé cuenta, quedará infectado y podrá acarrear muchos problemas.

2. OBJETIVOS

Para lograr las metas fijadas en el desarrollo del proyecto se han planteado los siguientes objetivos de trabajo.

- **Objetivo general:**

Como se ha introducido anteriormente, el objetivo es ayudar a tomar conciencia para evitar los ataques de ingeniería social canalizados mediante USB´s. Estudiaremos los ataques a los que podríamos estar expuestos casi diariamente pudiendo comprobar los daños que se pudieran ocasionar en el caso de que fuésemos infectados.

Trataremos de dar a conocer un gran problema del que casi no recibimos educación y, de este modo, intentaremos luchar contra casos como estos. Con ese fin, se va a hacer pública una plataforma para descubrir el nivel de concienciación de una institución dada. Esto es, los elementos necesarios para llevar a cabo una campaña desde que se dejan los USB infectados hasta la visualización de los resultados sobre los cuales tomar las correspondientes decisiones.

- **Objetivos específicos:**

1. Conocer lo que es la ingeniería social. Se detallará de que se trata y para que se utiliza, así como sus diferencias metodológicas de uso y cómo prevenir y actuar contra esta.

2. Veremos los tipos de ataque que se pueden emplear desde los USB y el impacto que pueden ocasionar.

3. También se intentará dar una visión del diseño de la aplicación, detallando las distintas interacciones entre los distintos módulos existentes con diagramas que muestran distintas transiciones.

4. Se detallará la implementación de la herramienta, esto es, el entorno de desarrollo tecnológico empleado, los diferentes lenguajes de programación usados y las estructuras.

5. Explicación del plan de pruebas llevado a cabo.

3. ESTADO DEL ARTE

3.1. Introducción y etapas

Según Sandoval Castellanos, E. J., “La Ingeniería Social es el acto de manipular a una persona a través de técnicas psicológicas y habilidades sociales para cumplir metas específicas. Éstas contemplan entre otras cosas: la obtención de información, el acceso a un sistema o la ejecución de una actividad más elaborada (como el robo de un activo), pudiendo ser o no del interés de la persona objetivo.”

La ingeniería social, por lo tanto, por el simple hecho de depender del uso del ser humano, se trata de una vulnerabilidad totalmente independiente de la plataforma tecnológica.

Cada ataque mediante este tipo de manipulación es único, pero se puede encontrar un ciclo de vida de la amenaza válido, teniendo en cuenta todas las actividades que afecta un proyecto de este tipo. La representación general podría ser:

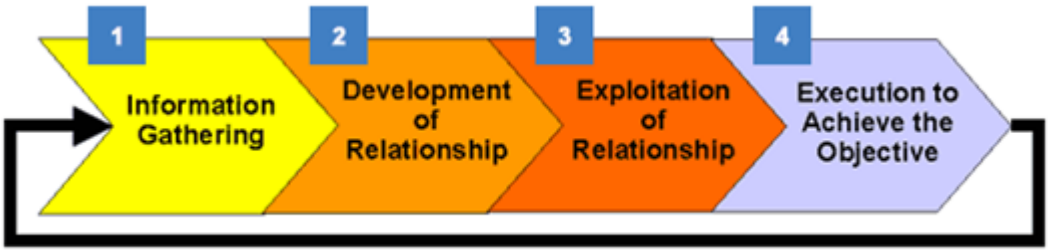


Figura 1: Etapa de los ataques de ingeniería social

1. **Footprinting:** es la primera etapa, una de las tareas realizadas antes de comenzar el ataque real. Se trata de una intrusión en la cual se acumula toda la información sobre los objetivos y el entorno de estos, con el fin de establecer una relación y así asegurar las posibilidades de éxito. La recopilación de información durante esta etapa no está limitada y cuánta más información se consiga acumular, el ataque se realizará con mayor precisión.

Algunas de las herramientas utilizadas durante esta fase son: *creepy, SET y Maltego.*

2. **Relación de confianza:** una vez alistados todos los objetivos, el atacante desarrolla una relación con el objetivo (generalmente se trata de un empleado) para ganarse su confianza. Esta confianza creada se utilizará posteriormente para conseguir datos confidenciales, y así causar daños graves al negocio del que ese primordial objetivo pertenece.

3. **Manipulación psicológica:** durante esta etapa, el ingeniero social utiliza la confianza que se ha ganado anteriormente para conseguir todos esos datos confidenciales que le interesan y así poder penetrar al sistema con muchísima más facilidad. Una vez recopilada la información requerida, el ingeniero social puede pasar al siguiente objetivo o avanzar hacia la exploración del sistema actual.

4. **Ejecución y salida:** finalmente, tras ejecutar todos los pasos anteriores y haber extraído toda la información conveniente, el ingeniero social tiene que hacer una salida que desvíe cualquier tipo de sospecha hacia él y asegurándose de que dejen existir huellas sobre su identidad y su vinculación al ataque.

3.2. Tipología de los ataques

Tras desglosar y explicar detalladamente cada una de las partes del ciclo de vida diseñado como prototipo, explicaremos a continuación cada uno de los tipos de ataques o técnicas que se utilizan dentro de la fase de Relación de confianza o de manipulación psicológica. Podemos dividirlos en dos grandes grupos:

- **Ataques locales:** este tipo de ataques son los que se realizan de forma física, es decir, sin necesidad de algún tipo de conexión de telecomunicaciones. Podemos diferenciar:

- **Pretexting (Suplantación):** es aquella técnica que se utiliza para conseguir ciertos datos o documentos llamando a la compañía o institución objetivo, utilizando una identidad falsa y con ciertos fines delictivos.
- **Tailgaiting:** el objetivo de este ataque es obtener acceso no autorizado a un área restringida mediante el engaño o descuido de una persona que si contiene esa autorización.
- **Dumpster Diving:** se trata de la búsqueda de papeles o documentos con información confidencial tras ser descartados o tirados a la basura.
- **Shoulder Surfing:** no se trata de otra cosa que de un espionaje "por encima del hombro", se observan los datos requeridos (claves de acceso al sistema, contraseñas, números PIN, etc.) sin que el objetivo se dé cuenta, por su espalda.
- **Misdirection (Distracción):** se trata de una técnica utilizada para despistar, llevando la atención de la víctima a algo irrelevante para captar la información requerida sin que se dé cuenta.
- **Baiting (Cebo):** para este tipo de ataques, generalmente se utilizan memorias USB con algún software malicioso. Los dejan "perdidos" cerca de la víctima que quieren atacar y para cerciorarse del éxito, se estudia a la víctima previamente, observando de forma más rápida la vulnerabilidad a explotar.

- **Ataques remotos:** estos tipos de ataques sí afectan a las redes de telecomunicación, incluida Internet. Los principales son:

  - **Phishing:** es el término más utilizado para estafar y obtener información confidencial de forma fraudulenta con el fin de causar pérdidas económicas a las víctimas (los datos robados suelen ser contraseñas o datos bancarios de cualquier tipo), se trata de una suplantación de identidad y puede llegar a ser muy peligrosa.
  - **Redes Sociales:** se trata de una técnica que cuenta con dos objetivos: por un lado, obtener información de la víctima, y por otro establecer una relación con la misma.

Actualmente, las redes sociales son los portales de Internet más utilizados, convirtiéndose en uno de los medios más sencillos para obtener información sobre las víctimas, ya que hay personas que cuentan lo que les ocurre de forma continua, a nivel tanto personal como profesional.

- **Telefónicos (Vishing):** en este caso, los atacantes utilizan encuestas a través de llamadas telefónicas para sacar información personal sin sospecha alguna.

3.3. Prevención y mitigación

Para completar este apartado introductorio e informativo sobre todo lo que rodea el mundo de la ingeniería social, trataremos algunos consejos para poder prevenir este tipo de ataques.

Algunos puntos fundamentales que se deben tener en cuenta son:

- **Administración de contraseñas:** al crear contraseñas debemos tener en cuenta una serie de pautas para conseguir que sean lo más robustas posibles: combinando caracteres, números, mayúsculas, minúsculas y caracteres especiales. Además, otro factor importante respecto a las contraseñas es la frecuencia con la que se deben cambiar, donde lo recomendado es cada seis meses.
- **Factores de autentificación múltiples:** si se encontraran sistemas con niveles de riesgo alto, es recomendable tener un mínimo de dos factores de autentificación.
- **Antivirus/Antiphising:** este tipo de software debe establecerse tanto en servidores finales como en las propias máquinas de cada usuario.
- **Administración de cambios:** un proceso de administración de cambios es importante tenerlo en cuenta, ya que puede minimizar los riesgos de ser atacados.
- **Clasificación de la información:** se debe hacer una clasificación de toda la información de la que disponemos, diferenciando la considerada como confidencial y no confidencial. Asimismo, es una tarea importante el

efectuar un manejo y tratamiento seguro de la información que se ha clasificado.

- **Manipulación y destrucción de registros:** es fundamental que todos los registros con datos de carácter sensible sean destruidos íntegramente, de tal modo que sea imposible volver a acceder a ellos.
- **Procesos operativos:** a todos aquellos procesos que definen cómo proveer acceso a información confidencial, hay que conservarlos bajo especial atención.
- **Seguridad física:** en este último apartado se recomienda la utilización de recursos como los sistemas biométricos o sistemas de video vigilancia, así como los requerimientos de acompañamiento para personas que no pertenezcan de forma directa a la institución.

A pesar de haber mencionado numerosos consejos previsores contra los ataques provenientes de ingenieros sociales, puede que no sea suficiente para garantizar la seguridad total. Sería conveniente, además, entrenar y concienciar a los usuarios sobre este aspecto, realizando auditorias y campañas de concienciación como la que se propone en este trabajo de fin de grado.

4. USB´S.

4.1. Introducción

El USB (Universal Serial Bus) es un dispositivo que se utiliza para el almacenamiento de información sin necesidad de baterías. El USB se ha convertido en uno de los sistemas de almacenamiento y transporte de datos más utilizado hoy en día, dejando prácticamente de lado a los antiguos sistemas de transporte y almacenamiento como podían ser los disquetes o CD's.

Actualmente se puede encontrar este tipo de dispositivos con mucha facilidad en el mercado y con diferentes tipos de capacidad, desde 1 GB hasta 256, siendo poco práctico a partir de los 64 GB por su alto coste, a pesar del pequeño espacio que ocupan.

4.2. Tipología de ataques mediante memorias USB

A continuación, vamos a proceder a explicar y clasificar los tipos de ataque que se pueden realizar mediante este tipo de dispositivos.

Entre todos los tipos de ataques que se pueden realizar mediante estas memorias, podemos realizar una clasificación de ellos como podemos observar a continuación:

A. **Ataques reprogramando el controlador del dispositivo USB:** este tipo de ataques reprograman el controlador del dispositivo USB, realizando un engaño del sistema. Al reprogramar el controlador, se realiza un cambio de cambio de funciones del dispositivo, llevando a cabo otro tipo de operación.

   En la *Figura 2*, que veremos al final de este punto de explicación, podemos ver algunos ejemplos de este tipo de ataques. Están marcados en rojo y numerados desde el punto 1 hasta el 9. Entre ellos podemos destacar el "*Rubber Ducky*", "*PHUKD*", "*USBdriveby*", "*Evilduino*" y el "*USB dongles*" (1, 2, 3, 4 y 8), los cuales se presentan como teclados e inyectan una secuencia de teclas precargadas configuradas en un script.

B. Este apartado globaliza a los **USB de tipo periférico**, pudiendo subdividirlo en dos grupos totalmente diferentes.

Para entender este grupo de clasificación, es necesario conocer el significado de '*firmware'*: se trata de un programa que reside en la memoria ROM y contiene una serie de instrucciones que permiten al hardware interactuar con el software. Por lo tanto, la función de este programa es la de recoger información y poner en marcha las peticiones del usuario facilitando su manejo.

B.1. Ataques reprogramando el firmware del dispositivo: en estos ataques, la reprogramación puede ser ejecutada mediante actualizaciones o haciéndose pasar por un proceso legítimo. Es decir, mediante la creación de un protocolo invisible para el usuario, que a simple vista parece no mostrar ningún tipo de cambio, mientras que toda la ejecución se realiza en segundo plano.

En la *Figura 2* podemos ver algunos ejemplos de este tipo de ataques. Están marcados en amarillo y numerados desde el punto 10 hasta el 17. Entre ellos podemos hacer mención a diferentes tipos como, por ejemplo, El "*ataque HID a Smarthphones*" (10), el cual crea un controlador malicioso para aparentar ser un teclado USB, el "*Virtual Machine Break-Out*" (15), los cuales usaban el firmware para poder realizar una evasión de los entornos de máquinas virtuales y el ataque "*iSeeYou*", que utilizaba la reprogramación del firmware para poder realizar captura de video de forma oculta y sin la necesidad de la notificación del indicador LED.

B.2. Ataques basados en dispositivos USB sin reprogramación: en este caso, no se utiliza la reprogramación del firmware, sino que se usan otro tipo de técnicas variadas entre las que destacan la ocultación de malware y la explotación de desbordamientos de buffer del sistema operativo entre otras.

En la *Figura 2* podemos ver algunos ejemplos de este tipo de ataques. Están marcados en azul y numerados desde el punto 18 hasta el 28. De entre ellos podemos mencionar el "*USB backdoor*" (19), el cual es uno de los que oculta malware, en este caso para guardar comandos

preestablecidos que convierten unidades de red en unidades lógicas, en redes aisladas. Otro claro ejemplo de este tipo de ataque puede ser el "*ataque a teléfonos inteligentes a través del puerto USB*" (27), que como en el caso anterior tienen la funcionalidad de ocultar y entregar malware, pero en este caso la finalidad seria engañar al usuario pensando que se trata de un simple cargador USB de dispositivos móviles.

C. **Ataques eléctricos:** estos ataques son los menos frecuentes, pero son los que mayor poder de destrucción tienen. En términos generales, siempre suelen ser dirigidos a personas específicas y el objetivo es dejar totalmente inutilizable el dispositivo victima a través de una descarga eléctrica.

   En la *Figura 2* podemos ver un ejemplo de este tipo de ataques. Está marcado en morado y numerado con el número 29. El nombre de este ataque es, "*USB killer*", y tiene como función emitir una descarga eléctrica de tal forma que destruya por completo el dispositivo y quede fuera de servicio.

4.3. Riesgos y prevención

Como hemos estado mencionando durante todo este capítulo, las memorias USB son un medio de transporte de datos muy común y muy utilizado hoy en día, sin embargo, un mal uso de estos puede suponer un serio riesgo para cualquier institución.

Los siguientes riesgos son los que más preocupan y a los que más atención se debería prestar:

- **Utilización de memorias USB como copias de seguridad**: aunque se puedan utilizar para ello, no dejan de ser un dispositivo físico y por ello corren el riesgo de poder extraviarse o sufrir algún tipo de avería que haga perder la información.

- **Sustitución de una red como medio de compartimiento**: no se debe dejar de utilizar la red como medio de compartición simplemente porque

dispongamos de memorias USB, ya que el sentido de una red es el de poder participar en la compartición de archivos de forma constante y, además si solo empleamos las memorias como medio de compartición podemos sufrir lo que hemos comentado en el punto anterior.

- **Sustracción de datos**: la sustracción de cualquier tipo de información que se disponga de una compañía, podría suponer una posible brecha de seguridad en caso de que la información extraída fuese de carácter confidencial.
- **Infección y propagación de virus**: éste es el principal riesgo y el que más concierne a las empresas. La introducción de un malware puede tener consecuencias muy graves para una empresa, ya sea por pérdida de información confidencial, por robo de identidades o por el costo que puede suponer en arreglo de infraestructura. No vale de nada tomar precauciones en cuanto a la navegación por Internet o a la fiabilidad de las descargas directas, si luego introducimos en nuestro ordenador una memoria USB que no es de nuestra propiedad, ya que el peligro puede ser igual o mayor que en los casos anteriores.
- **Físicos**: como hemos mencionado previamente, al ser un dispositivo físico puede sufrir problemas de deterioro o problemas a la hora de conectar debido al dañado de cables que causarían la pérdida de la información depositada en ellos.

Para poder prevenir estos ataques la mejor medida que se puede tomar es la concienciación y el entrenamiento a personas con respecto a estos, aunque también se han adoptado medidas como:

- **Inhabilitación del uso de los puertos USB**: en caso de necesitar la realización de una transferencia de datos se haría uso de la red interna.
- **Software actualizado**: con el fin de evitar cualquier tipo de exposición a ser vulnerable.
- **Analizar las memorias USB con el antivirus**: para poder comprobar que no existe ningún tipo de riesgo de infección que pueda poner en compromiso a la compañía.

- **No utilizar memorias USB en ordenadores con origen desconocido**: las memorias se han de utilizar en ordenadores confiables y a ser posible de forma interna, de tal forma que no haya ninguna posibilidad de intervención de terceros y por lo tanto la única dependencia sea nuestra.
- **No excederse en el uso de las memorias USB para almacenar**: existe una gran variedad de soluciones con mayor seguridad, fiabilidad y flexibilidad a la hora de realizar copias de respaldo de forma metódica, entre las que podemos destacar los servicios en la nube o los servidores externos.

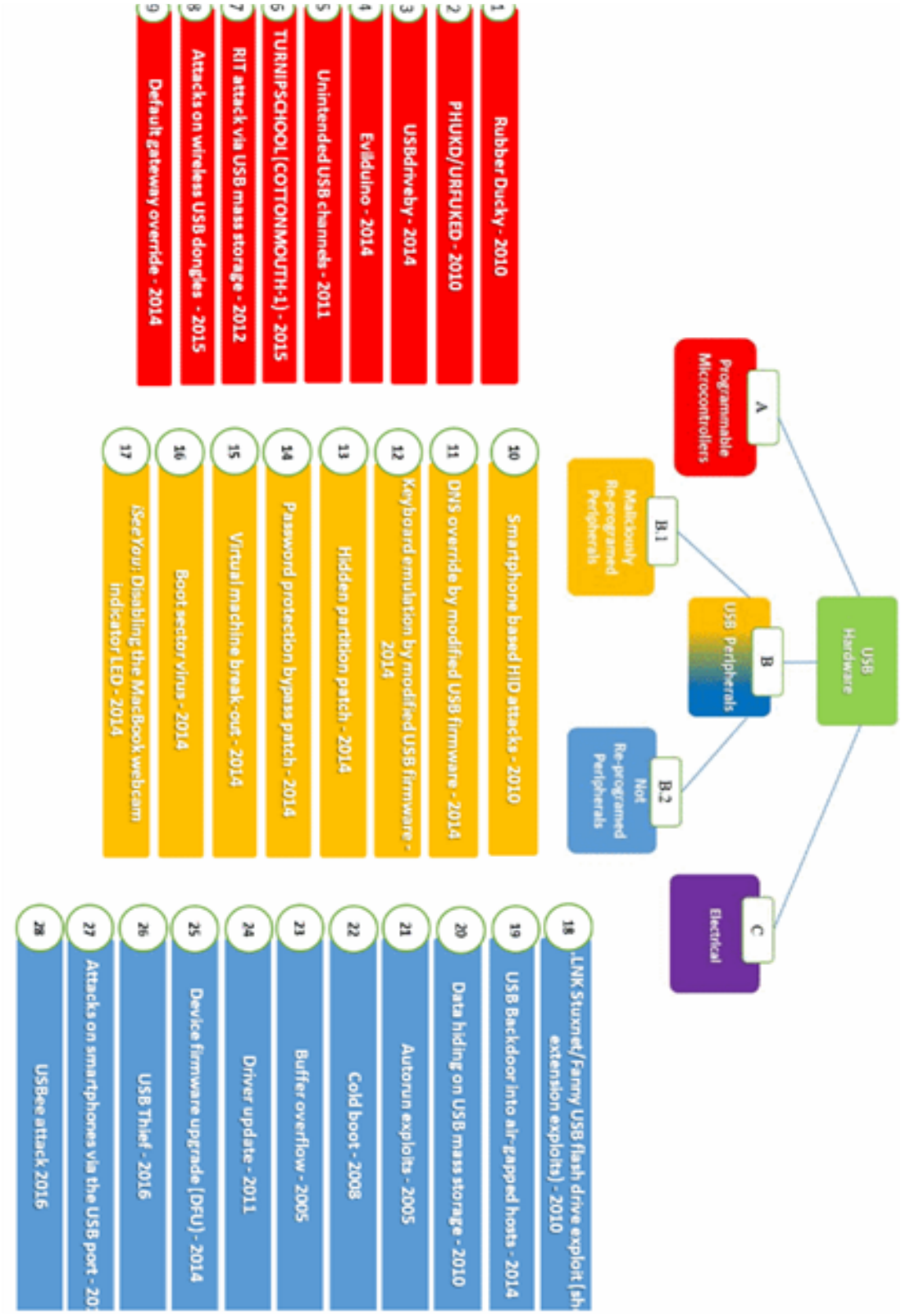


Figura 2: Clasificación de los ataques mediante memorias USB

5. ARQUITECTURA

5.1. Entorno tecnológico

Para el funcionamiento adecuado de la plataforma diseñada es necesario que los equipos en los cuales se ejecuta, tanto dicha aplicación como el servidor, tienen que verificar unos mínimos requisitos.

Uno de estos requisitos es la conexión de la aplicación a un servidor, al cual podemos acceder de dos maneras: mediante una conexión a Internet, o bien, mediante una misma red local en el caso de que ambos (aplicación y servidor) estén en el mismo sistema.

Por lo tanto, conectarse a internet no requiere grandes requisitos de hardware. Para una configuración local en nube, por ejemplo, bastaría con una utilización de 32 MB de RAM, aunque es aconsejable una mayor capacidad de memoria para poder garantizar un mejor rendimiento. El procesador, por otro lado, tampoco requiere una limitación, aunque lo adecuado sería utilizar un mínimo de 500MHz para garantizar un tiempo de respuesta satisfactorio.

5.2. Flujo de interacción

Se ha diseñado un sistema que consta de tres componentes: una aplicación, un servidor colector de datos y una herramienta para el acceso visual de los datos.

1. **Aplicación:**

Se ha realizado un programa en lenguaje C#, que se encarga de sustraer los datos de las máquinas que intentan acceder al archivo ejecutable.

La finalidad del programa es la de extraer ciertos datos de las “maquinas víctimas”, con el fin de poder identificar a las personas que han “caído” en el engaño y así poder hacer la campaña de concienciación necesaria.

Para cada campaña se van a configurar un cierto número de USB´s, a definir por la empresa interesada. Al dar de alta al cliente, se le solicitará que introduzcan

el número de memorias maliciosas que quieren configurar, de manera que se configurará cada memoria con un identificador único.

El identificador del USB y la campaña propia a la cual se hace referencia se asocian a un fichero conf.xml, pudiendo sustraer de cada máquina datos de entre los siguientes:

- Nombre de la máquina (Machname).
- Nombre del usuario (userName).
- Versión del sistema operativo (osVersion).
- Fecha y hora (netInfo).

En cada USB se inserta el archivo conf.xml de modo oculto para que no lleve a sospecha y el propio archivo ejecutable camuflado, del cual hablaremos más adelante.

Los datos sustraídos se envían a un colector de datos utilizando el protocolo de aplicación HTTP. Sin embargo, los datos que se transmiten mediante este protocolo viajan en texto plano y por tanto seria vulnerable a ataques de tipo MITM (Man In The Middle), en donde un atacante podría introducirse en medio de la comunicación entre la aplicación y el servidor y robarnos los datos. Para solucionar esto, se han cifrado los datos sustraídos aplicando el algoritmo de cifrado de clave simétrica *3DES (TripleDES)* antes de enviarlos, de tal forma que si alguien intenta interceptarlos no tendrá visibilidad legible sobre los datos.

2. **Servidor colector de datos:**

Se dispone de un servidor en la nube, en el cual se ha configurado una base de datos Mysql, donde se van a almacenar de forma ordenada los datos recogidos por cada memoria USB. En el anexo B figura como se ha realizado la instalación y acceso al servidor.

Como hemos comentado previamente, los datos se envían cifrados a través del protocolo HTTP y se envían directamente a un receptor denominado info.php, residente en el servidor, el cual es el encargado de recopilar los datos sustraídos, aplicarles el descifrado e insertarlos correctamente en la base de datos.

3. **Herramienta:**

Finalmente, tenemos una herramienta en la cual poder observar los resultados a tiempo real. Para ello, se han definido previamente dos roles: administrador y usuario.

Se dispone de un solo rol de administrador, el cual tiene los privilegios de dar de alta nuevas empresas para realizar las campañas, así como de visualizar los resultados de las campañas de cada empresa. Cabe mencionar que cada campaña va asociada a una empresa, de tal forma que si una empresa desea hacer varias campañas habría que registrarlas tantas veces como campañas se desean realizar.

El rol de usuario tiene capacidad de observar el número de USB´s que se han configurado, los resultados que se van obteniendo a tiempo real de la campaña y también puede ver los resultados de forma estadística, sabiendo de manera exacta que USB´s han sido los que se han pulsado. Esto último es interesante, ya que si se configuran los USB´s y se dejan en diferentes sitios de forma estratégica, tendremos la posibilidad de saber cuáles han sido las zonas que han tenido más éxito.

6. DESARROLLO E IMPLEMENTACIÓN

6.1. Equipo de desarrollo

A continuación, se exponen las características tanto hardware como software del equipo sobre el cual se ha llevado a cabo el desarrollo de la aplicación.

Hardware:

El proyecto se ha desarrollado en un equipo con las siguientes características:

Equipo portátil Lenovo de 64-bits, con disco duro de 250 GB, sistema operativo Windows 10 y procesador Intel Core I5-7200U de 2.5GHz.

Software:

Para la codificación e implementación del proyecto, se ha usado el siguiente software:

Software de programación:

- VisualStudio para la codificación de las memorias USB.
- Control de versiones mediante GitHub en Windows.
- MySQL Workbench para ver los contenidos de la base de datos.

Software de edición:

- Día.
- Sublime Text para edición de texto.
- Photoshop para edición imágenes.

6.2. Plataformas y lenguajes de programación

A continuación, se nombran y se explican los lenguajes de programación y plataformas utilizadas en el proyecto:

HTML

HTML es un lenguaje de marcado que se utiliza para el desarrollo de páginas de Internet.

Es el elemento de construcción más básico de una página web y es utilizado para la creación y representación visual de una página web.

A lo largo de este proyecto, he utilizado este lenguaje para dar forma y amoldar toda la interfaz gráfica de la plataforma.

Servidor en AWS

Amazon Web Services (AWS) es una plataforma de servicios en la nube. En concreto se ha utilizado EC2 (Elastic Compute Cloud), el cual elimina la necesidad invertir inicialmente en hardware y permite lanzar tantos servidores virtuales como necesite, configurar la seguridad y las redes, y administrar el almacenamiento.

En este proyecto, el servidor tiene un sistema operativo Ubuntu 16.04, y se ha configurado el servidor de forma que solo se permita el tráfico a través de los protocolos HTTP, SSH, HTTPS y MYSQL.

MySQL

MySQL es un sistema de gestión de bases de datos relacional, y considerado como una de las bases de datos open source más populares, sobre todo para entornos de desarrollo web.

La base de datos de usuarios, que contiene tanto los nombres de usuarios como sus contraseñas está implementada en MySQL. Para la conexión del servidor con la base de datos, se ha realizado a través del puerto 127.0.0.1 (Localhost).

XML

XML (eXtensible Markup Language), es un lenguaje de marcas desarrollado utilizado para almacenar datos en forma legible. Su utilidad reside en que permite estructurar documentos grandes, además de dar soporte a bases de datos, y ser compatible entre sistemas para compartir la información de una manera segura, fiable y fácil.

En este proyecto lo hemos utilizado para la configuración de parámetros a sustraer en las diferentes campañas.

PHP

Se trata de un lenguaje de código abierto de uso general de código del lado del servidor. Este lenguaje se utiliza para generar páginas dinámicas, de tal forma que está continuamente interactuando con la base de datos.

En este proyecto la mayor parte de las páginas han sido desarrolladas en este lenguaje, incrustado en el lenguaje HTML mencionado anteriormente, lo que nos ha facilitado la visualización del contenido de la base de datos en una interfaz gráfica.

C#

Se trata de un lenguaje de programación orientado a objetos y en el cual se sigue una sintaxis básica derivada de los lenguajes C y C++, con la diferencia de que en este caso utiliza un modelo de objetos de la plataforma .NET (Microsoft), muy similar al de Java.

En este proyecto hemos utilizado este lenguaje para la programación del archivo ejecutable.

CSS

Se trata de un lenguaje que se utiliza para definir la apariencia de documentos estructurados escritos en un lenguaje marcado, como puede ser HTML.

En este proyecto hemos utilizado este lenguaje para el diseño y apariencia de nuestra aplicación.

JavaScript

Se trata de un lenguaje de programación que se utiliza del lado del cliente y permite la creación de efectos atractivos y dinámicos en las páginas web.

En este proyecto hemos utilizado este lenguaje para acompañar al lenguaje CSS y realizar efectos dinámicos en varios sitios diferentes de nuestra aplicación, como por ejemplo los menús desplegables o los checkboxes.

6.3. Estructuración del código

A continuación, se va a mostrar los diferentes módulos creados para cada uno de los componentes.

6.3.1. Aplicación

TripleDES.cs: este módulo es el encargado de encriptar y desencriptar los datos aplicando el algoritmo simétrico TripleDES, en el cual hemos hecho uso de la librería System.Security.Cryptography.

SystemInfo.cs: este módulo se encarga de configurar los datos a sustraer de la máquina, así como de utilizar el módulo TripleDES para cifrar el contenido sustraído, y también se encarga de enviar los datos cifrados al servidor a través del protocolo HTTP.

Program.cs: este módulo es el main de nuestro programa y se encarga de hacer la llamada al módulo SystemInfo.

Conf.xml: archivo en el cual se va a establecer el identificador de usuario y el identificador de la campaña, así como los parámetros a sustraer.

6.3.2. Servidor colector de datos

Phishing_campaigninfo.sql: script encargado de crear la tabla que se encargará de almacenar los datos sustraídos de cada memoria USB.

Phishing_userinfo.sql: script encargado de crear la tabla que se encargará de almacenar los datos correspondientes a los usuarios de la plataforma.

6.3.3. Herramienta

Archivos CSS:

- **Fg_membersite.css, pwdwidget.css, style.css, style2.css, style3.css, bootstrap.css, custom.css y Font-awesome.css**: estos son los ficheros CSS que proporcionan la apariencia de la interfaz gráfica de la herramienta.

Archivos de JavaScript:

- **Index.js, index2.js, bootstrap.js, custom.js y jquery-1.10.2.js**: estos son los ficheros JavaScript que he utilizado para proporcionar animación y mejorar el diseño y apariencia de la aplicación.

Archivo HTML:

- **Blank.html**: se trata de un fichero de plantilla, que muestra un ejemplo de cómo sería una posible configuración de los settings.

Archivo PHP:

- **CachedPDOStatement.php**: fichero utilizado para reportar posibles errores que surjan con respecto a la conexión y gestión de la aplicación con la base de datos.
- **Password.php**: fichero que gestiona la verificación de contraseñas y les aplica una función hash, de tal forma que no se guarden en texto plano y se disponga de una mayor seguridad.
- **Class.user.php**: fichero que se encarga de realizar las funciones de registro, login, redireccionamiento y logout. Para ello requiere del fichero anterior, password.php, para poder validar las contraseñas.

- **Db.php**: fichero que se encarga de establecer la conexión con la base de datos. Para ello incluye y evalúa la ejecución del fichero anterior, Class.user.php, de tal forma que se puedan evaluar las funciones mencionadas anteriormente.
- **Index.php**: fichero de inicio de sesión, que requiere del fichero anterior Db.php, para poder establecer una conexión con la base de datos y comprobar si el usuario tiene acceso de usuario o de administrador y poder redirigirle a su correspondiente Dashboard.
- **AdminDashboard.php**: fichero que muestra el menú y las diferentes opciones que posee un usuario con roles de administrador.
- **UserDashboard.php**: fichero que muestra el menú y las diferentes opciones que posee un usuario.
- **Info.php**: fichero encargado de recoger los datos sustraídos mediante el archivo ejecutable y de insertarlos correctamente en la base de datos.
- **Settings.php**: fichero que muestra una plantilla estática para ilustrar de una posible página en donde se pudiesen ajustar los diferentes parámetros.
- **Edit.php**: fichero que muestra una plantilla estática para ilustrar de una posible página en donde se pudiesen editar los diferentes parámetros.
- **Campaigns.php**: fichero que muestra el id de la campaña y el número de memorias USB que han sido ejecutadas.
- **Campaign_view.php**: fichero que muestra un nuevo Dashboard en el cual se nos muestra tanto el nombre de la compañía como el número de memorias USB ejecutadas, y nos da la posibilidad de ver los resultados de la campaña, así como de las estadísticas.
- **Newindex.php**: fichero que muestra todos los resultados de la campaña, en donde podemos ver el id de la memoria USB, la hora, el nombre de la máquina, el nombre de usuario y la versión del sistema operativo.
- **Print_results.php**: fichero que nos muestra las estadísticas de las diferentes campañas.
- **Sign-up.php**: fichero para poder dar de alta un nuevo usuario en la aplicación.

6.4. Desarrollo y funcionalidad del sistema

A continuación, se va a detallar como se ha desarrollado cada componente del sistema:

- **Aplicación:**

El programa ejecutable se divide en cuatro ficheros como hemos mencionado en el apartado 5.3.2.

El fichero program.cs, es el main principal que simplemente realiza una llamada al archivo SystemInfo.cs, que es el que se encarga de realizar toda la funcionalidad como veremos a continuación. En la *Figura 3*, podemos observar que tenemos una función denominada gatherEncryptedInfo, ésta es la encargada de recoger los datos de las máquinas y guardarlos encriptados.

```
private void gatherEncryptedInfo()
{

    XmlDocument xmlDoc = new XmlDocument(); // Create an XML document object
    xmlDoc.Load("conf.xml"); // Load the XML document from the specified file

    // Get elements

    this.date = this.tripleDES.EncryptData(DateTime.Now.ToString());
    if (xmlDoc.GetElementsByTagName("machineName")[0].InnerText == "1")
        this.MachineName = this.tripleDES.EncryptData(Environment.MachineName);
    if (xmlDoc.GetElementsByTagName("userName")[0].InnerText == "1")
        this.UserName = this.tripleDES.EncryptData(Environment.UserName);
    if (xmlDoc.GetElementsByTagName("netInfo")[0].InnerText == "1")
        this.networkInfo = this.tripleDES.EncryptData(this.getNetworkInfo());
    if (xmlDoc.GetElementsByTagName("osVersion")[0].InnerText == "1")
        this.OsVersion = this.tripleDES.EncryptData(Environment.OSVersion.ToString());
    this.campaignID = this.tripleDES.EncryptData(xmlDoc.GetElementsByTagName("campID")[0].InnerText);
    this.usbID = this.tripleDES.EncryptData(xmlDoc.GetElementsByTagName("usbID")[0].InnerText);


}
```

Figura 3: Función gatherEncryptedInfo, encargada de sustraer los datos de las máquinas

Como podemos ver en la imagen, para recoger los datos de la maquina utilizamos la variable "*Environment*" de C#, y una vez recogidos los datos llamamos a la función EncryptData que reside en el archivo TripleDES, del que hablaremos a continuación. Tras haber recogido y guardados los datos,

procedemos a enviarlos al servidor y para ello utilizamos la función sendData, en donde declaramos un Webclient y enviamos los datos a través del protocolo HTTP al archivo info.php encargado de recolectar la información.

En cuanto al archivo TripleDES, se tienen dos funciones EncryptData y DecryptData, que son las encargadas tanto de cifrar como de descifrar la información aplicando el algoritmo TripleDES y utilizando la clave que hemos definido con el valor de "*holamundo*" para ello.

Por otro lado, tenemos un archivo conf.xml, como podemos observar en la *Figura 4*, en donde el primer parámetro hace referencia al identificador que se le asigna a ese dispositivo USB, y el segundo parámetro se trata del identificador de la compañía que realiza la campaña. El resto de los parámetros son configurables, de tal forma que si se decide que se recojan hay que ponerles un valor de 1, mientras que, si queremos hacer una excepción y omitir alguno de los parámetros, simplemente debemos poner un 0.

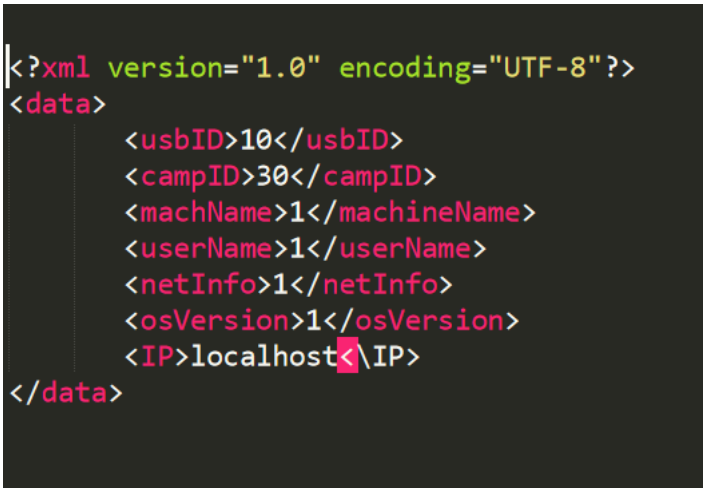


Figura 4: Archivo Conf.xml, encargado de seleccionar los datos a sustraer

A la hora de configurar las memorias USB, habrá que incluir tanto el archivo ejecutable en sí como el archivo conf.xml, de tal forma que tengamos identificado de que dispositivo se trata. El archivo conf.xml lo pondremos de forma oculta, para que no sea visible a los usuarios y así evitemos crear cualquier tipo de duda o confusión, por lo que simplemente habrá que dar click derecho sobre el archivo e ir a la pestaña properties y en atributes marcar la opción hidden.

Una vez ocultado el archivo conf.xml, pasaremos a camuflar nuestro archivo ejecutable. Para ello hay diferentes metodologías, de entre las que destacan las siguientes:

RTLO method

Utilizaremos el programa *character map* para copiarnos el carácter especial ASCII, *U+202E*, que efectúa una anulación de derecha a izquierda con el fin de hacer que el ejecutable aparente ser un documento PDF legítimo. Por lo que si por ejemplo utilizamos el siguiente nombre para el fichero “ExamenesInd fdp.exe” e introducimos el carácter especial entre Ind y fdp, obtendremos como resultado “ExamenesIndexe.pdf” y por lo tanto quedará reflejado como si fuese un archivo de extensión pdf.

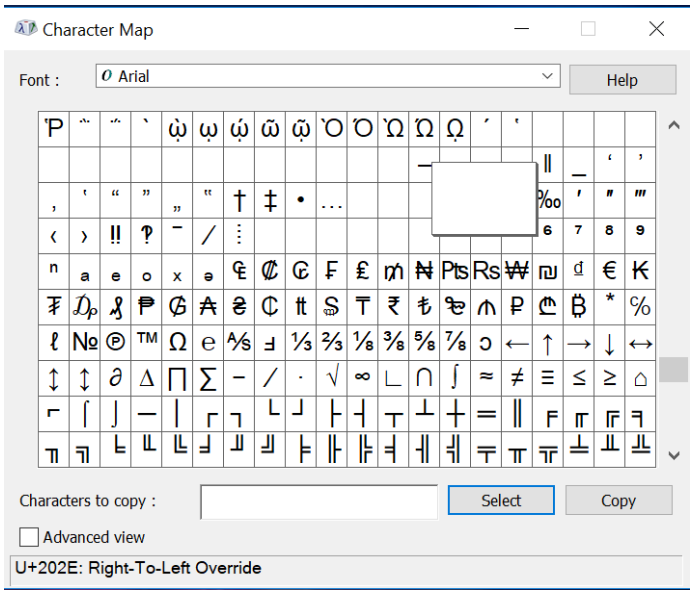


Figura 5: Carácter especial U+202E

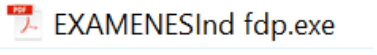


Figura 6: Nombre inicial del archivo ejecutable

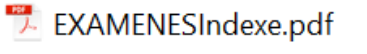


Figura 7: Nombre camuflado

Aplicación de espacios

En este caso, dejaremos la extensión del formato en .exe y simplemente lo que haremos es insertar una cantidad considerable de espacios, de tal forma que se oculte la extensión del fichero y solo aparezcan visibles los tres puntos suspensivos como se refleja en la imagen a continuación.

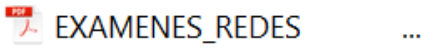


Figura 8: Extensión oculta mediante espacios

Acceso directo al archivo original

En este caso, lo que haremos es ocultar el archivo ejecutable original y sacar un acceso directo a él, en el cual podemos aplicar el nombre y la extensión que nos plazca. Para ello, realizamos click derecho en el archivo ejecutable original y seleccionamos la opción de shortcut, la cual nos sacará el acceso directo del archivo al que nombraremos como deseemos. Una vez realizado esto, solo quedaría ocultar el archivo ejecutable original, por lo que haremos clic derecho en el archivo y en properties seleccionaremos la opción hidden como en el caso del conf.xml. A continuación, se muestra un ejemplo de cómo quedaría.

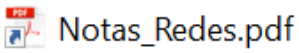


Figura 9: Acceso directo al archivo ejecutable original

- **Servidor colector de datos:**

En el caso del servidor de datos, se puede encontrar la configuración y acceso en el anexo B. Una vez instalado y configurado, procedemos a crear la base de datos. Por lo que ingresamos en MySQL mediante el comando "*mysql -u nombre de usuario -p password"*, y una vez dentro, creamos la base de datos,

a la que he denominado phishing, mediante el siguiente comando "*CREATE DATABASE phishing"*. Una vez tenemos la base de datos creada, se dispone de dos scripts para crear las dos tablas en las que se van a almacenar los datos recogidos de forma ordenada como se muestra a continuación.

```
DROP TABLE IF EXISTS `campaigninfo`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `campaigninfo` (
  `id_usb` varchar(50) DEFAULT NULL,
  `Hour` varchar(45) DEFAULT NULL,
  `machineName` varchar(45) DEFAULT NULL,
  `userName` varchar(45) DEFAULT NULL,
  `osVersion` varchar(45) DEFAULT NULL,
  `id_comp` varchar(45) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `campaigninfo`
--

LOCK TABLES `campaigninfo` WRITE;
/*!40000 ALTER TABLE `campaigninfo` DISABLE KEYS */;
INSERT INTO `campaigninfo` VALUES ('1','2017.11.21','LES001611','moreno','Microsoft Windows NT 6.2.9200.0','30'),
/*!40000 ALTER TABLE `campaigninfo` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;
```

Figura 10: Script para la creación de la tabla correspondiente a las campañas

```
DROP TABLE IF EXISTS `userinfo`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `userinfo` (
  `userName` varchar(100) NOT NULL,
  `pass` varchar(500) DEFAULT NULL,
  `compName` varchar(45) DEFAULT NULL,
  `role` varchar(45) DEFAULT NULL,
  `id_comp` int(11) NOT NULL AUTO_INCREMENT,
  `usbCounter` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`id_comp`,`userName`),
  UNIQUE KEY `id_comp_UNIQUE` (`id_comp`)
) ENGINE=InnoDB AUTO_INCREMENT=8 DEFAULT CHARSET=utf8;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `userinfo`
--

LOCK TABLES `userinfo` WRITE;
/*!40000 ALTER TABLE `userinfo` DISABLE KEYS */;
INSERT INTO `userinfo` VALUES ('admin','8c6976e5b5410415bde908bd4dee15dfb167a9c873fc4bb8a81f6f2ab448a918','UAM','admin',1,NULL),
/*!40000 ALTER TABLE `userinfo` ENABLE KEYS */;
UNLOCK TABLES;
```

Figura 11: Script para la creación de la tabla correspondiente a los usuarios del sistema

Para poder cargar las tablas correctamente, primeramente, debemos ingresar en la base de datos en la que se quieren incluir estas tablas, en nuestro caso la base de datos phishing, mediante el comando "*USE phishing*", y una vez dentro de nuestra base de datos, simplemente tendremos que ejecutar el siguiente código para cada script, "*SOURCE script.sql*" (En donde sustituiremos script.sql por el nombre de nuestros scripts).

Tras ejecutar los scripts ya dispondremos de nuestra base de datos con las correspondientes tablas, las cuales podremos observar utilizando el comando "*SHOW tables*", y podremos observar el contenido de cada una lanzando las siguientes querys "*Select * from userinfo*;" y "*Select * from campaigninfo*;" de tal forma que obtendremos una imagen visual como a continuación:

```
mysql> select * from campaigninfo;
+--------+---------------------+-------------+----------+-----------------------------------------+---------+
| id_usb | Hour                | machineName | userName | osVersion                               | id_comp |
+--------+---------------------+-------------+----------+-----------------------------------------+---------+
| 1      | 2017.11.21          | LES001611   | moreno   | Microsoft Windows NT 6.2.9200.0         | 30      |
| 1      | 2017.11.21          | LES001611   | morenos  | Microsoft Windows NT 6.2.9200.0         | 30      |
| 1      | 2017.11.21          | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 30      |
| 3      | 2017.11.21          | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 5       |
| 3      | 2017.11.21          | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 5       |
| 10     | 2017.11.21          | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 7       |
| 10     | 21/11/2017 11:41:44 | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 7       |
| 10     | 21/11/2017 13:17:38 | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 7       |
| 10     | 21/11/2017 14:28:00 | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 7       |
| 10     | 21/11/2017 14:49:44 | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 7       |
| 10     | 21/11/2017 18:35:45 | LES001611   | mmellouk | Microsoft Windows NT 6.2.9200.0         | 7       |
| 20     | 26/09/2018 17:11:18 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 9       |
| 20     | 26/09/2018 17:14:34 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 9       |
| 1      | 02/10/2018 15:01:23 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 02/10/2018 15:02:52 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 02/10/2018 15:12:10 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 02/10/2018 15:22:51 | LES004296   |          | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 02/10/2018 15:23:07 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 02/10/2018 16:28:20 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 02/10/2018 16:29:04 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 03/10/2018 15:12:00 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 13      |
| 4      | 03/10/2018 15:15:32 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 5       |
| 1      | 03/10/2018 16:03:55 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 03/10/2018 16:04:13 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 16      |
| 2      | 03/10/2018 16:07:31 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 16      |
| 1      | 04/10/2018 11:32:52 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 13      |
| 1      | 04/10/2018 11:40:35 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 13      |
| 3      | 04/10/2018 11:42:14 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 16      |
| 4      | 04/10/2018 11:43:07 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 16      |
| 1      | 04/10/2018 12:41:16 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 13      |
| 6      | 04/10/2018 13:08:56 | LES005337   | javiabal | Microsoft Windows NT 6.2.9200.0         | 16      |
| 7      | 04/10/2018 14:08:31 | LES001669   | esmoraur | Microsoft Windows NT 6.1.7601 Service Pack 1 | 16      |
+--------+---------------------+-------------+----------+-----------------------------------------+---------+
32 rows in set (0.00 sec)
```

Figura 12: Datos correspondientes a la tabla campaigninfo

Como podemos apreciar en la *Figura 12*, para la tabla campaigninfo hemos creado las siguientes columnas:

- **Id_usb**: en donde almacenaremos los identificadores de los USB´s configurados.
- **Hour**: para recoger la hora a la que se ha pulsado el USB.
- **machineName**: nombre de la máquina extraída.
- **userName**: nombre del usuario del equipo extraído.
- **id_comp**: identificador de la empresa en la cual se está realizando la campaña.
- **usbCounter**: número de USB´s configurados para la campaña.

```
mysql> select * from userinfo;
+------------------+------------------------------------------------------------------+-----------+-------+---------+------------+
| userName         | pass                                                             | compName  | role  | id_comp | usbCounter |
+------------------+------------------------------------------------------------------+-----------+-------+---------+------------+
| admin            | 8c6976e5b5410415bde908bd4dee15dfb167a9c873fc4bb8a81f6f2ab448a918 | Capgemini | admin |       1 |          0 |
| test@endesa.com  | 93fa3e4624676f2e9aa143911118b4547087e9b6e0b6076f2e1027d7a2da2b0a | Endesa    | user  |       5 |         17 |
| test@repsol.com  | 93fa3e4624676f2e9aa143911118b4547087e9b6e0b6076f2e1027d7a2da2b0a | Repsol    | user  |       7 |          4 |
| wai@wai.wai      | deccea0090575cf17422c91eedd8904fd8d927d30be3c25a245a73ce13fb9b05 | wai       | user  |       9 |          0 |
| nowai@nowai.wai  | deccea0090575cf17422c91eedd8904fd8d927d30be3c25a245a73ce13fb9b05 | nowai     | user  |      13 |          0 |
| test@prosodie.com | 93fa3e4624676f2e9aa143911118b4547087e9b6e0b6076f2e1027d7a2da2b0a | Prosodie  | user  |      16 |         12 |
+------------------+------------------------------------------------------------------+-----------+-------+---------+------------+
6 rows in set (0.00 sec)
```

Figura 13: Datos correspondientes a la tabla userinfo

Como podemos apreciar en la *Figura 13*, para la tabla userinfo hemos creado las siguientes columnas:

- **userName**: nombre del usuario del equipo para el acceso a la herramienta.
- **pass**: en donde almacenaremos las passwords de cada compañía, previamente hasheadas.
- **compName**: nombre de la empresa.
- **role**: rol de administrador o de usuario para acceder a la plataforma.
- **id_comp**: identificador de la compañía en la cual se está realizando la campaña.
- **usbCounter**: número de USB´s configurados para la campaña.

Finalmente, para concluir este apartado se muestra en la *Figura 14*, el fichero encargado de recoger los datos transmitidos por las memorias USB y de almacenarlos en la base de datos.

```php
$key = "holamundo";
$host = '127.0.0.1';
$db   = 'phishing';
$user = 'root';
$pass = 'Myrootpass';
$charset = 'utf8';

$dsn = "mysql:host=$host;dbname=$db;charset=$charset";
$opt = [
    PDO::ATTR_ERRMODE            => PDO::ERRMODE_EXCEPTION,
    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
    PDO::ATTR_EMULATE_PREPARES   => false,
];

$pdo = new PDO($dsn, $user, $pass, $opt);

//$machNameRes = $_POST['machineName'];

$date = decrypt($_POST['date'],$key);
$machNameRes =  decrypt($_POST['machineName'],$key);
$idUsb = decrypt($_POST['idUsb'],$key);
$userNameRes = decrypt($_POST['userName'],$key);
$osVersionRes = decrypt($_POST['osVersion'],$key);
$idCampaignRes = decrypt($_POST['idCampaign'],$key);
```

Figura 14: Descifrado de la información enviada por las memorias USB

Para realizar dicha función, como podemos ver en la *Figura 14*, el procedimiento es desencriptar los datos recibidos de las memorias USB utilizando la misma clave que se utilizó para cifrar los datos, e introducirlos en la base de datos mediante la query Insert que podemos ver en la siguiente imagen.

```php
$data_insert = sprintf("INSERT INTO campaigninfo (id_usb, id_comp, Hour, machineName, userName, osVersion) VALUES ( %s, '%s', '%s', '%s', '%s', '%s');"
    , $idUsb,  $idCampaignRes, $date, $machNameRes, $userNameRes, $osVersionRes);


$stmt = $pdo->query($data_insert);
```

Figura 15: Query para insertar la información en la base de datos

- **Herramienta:**

Se ha creado un login principal en el que, tras completar los datos de usuario y contraseña, nos redirigirá a un determinado *Dashboard*.

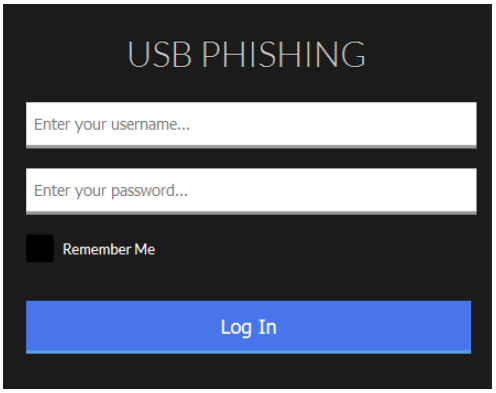


Figura 16: Pantalla de login de la herramienta

A continuación, vamos a dividir el apartado en dos subapartados, administrador y usuario, de tal forma que se puedan diferenciar los privilegios de uno y otro.

Administrador

Si ingresamos en el login con el rol de administrador, éste nos redirigirá al correspondiente Dashboard como podemos observar en la *Figura 17*.

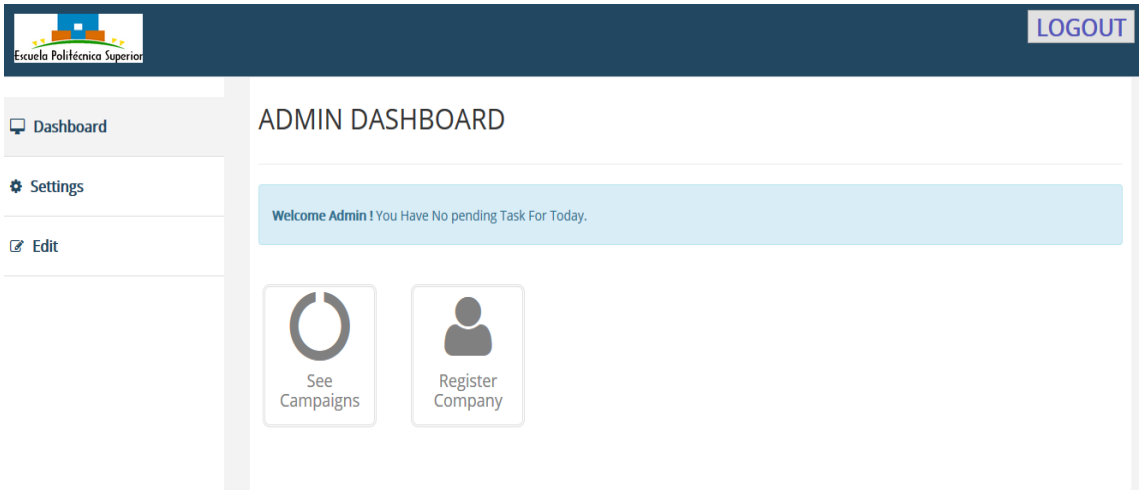


Figura 17: Dashboard correspondiente al rol administrador

En el cual podemos observar que se dispone de dos posibles acciones: See Campaigns y Register Company.

Si accedemos al apartado de See Campaigns, tendremos visibilidad de todas las campañas que se han llevado, o se están llevando a cabo, en las diferentes empresas, así como de los resultados de cada una de ellas. En la *Figura 18*, podemos ver todas las campañas que se han creado hasta el momento.

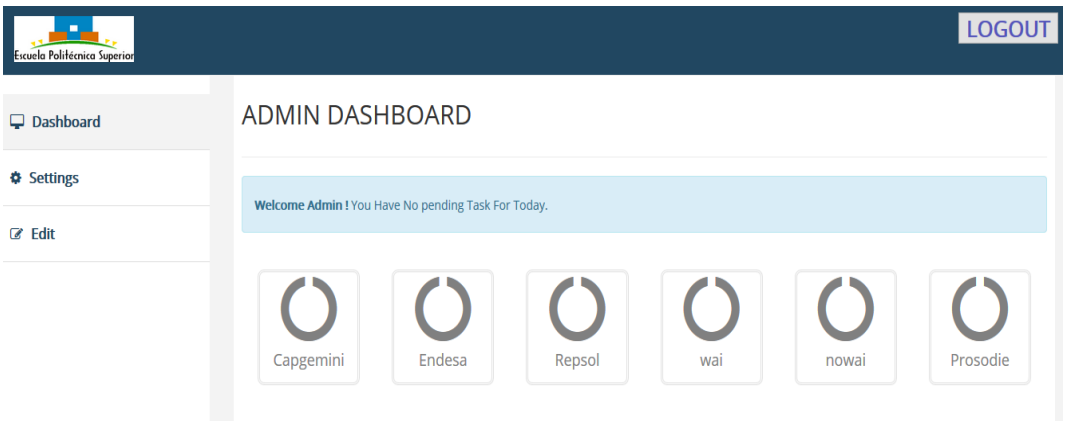


Figura 18: Total de campañas registradas en la herramienta

En caso de querer registrar una compañía, simplemente debemos tomar la opción de Register Company y nos mostrará un formulario, como el que se muestra en la *Figura 19*, en donde deberemos ingresar un nombre para la empresa, un email, el número de USB´s que se quieren configurar para la campaña y una contraseña. Una vez realizado, simplemente tenemos que pinchar el botón “create my Account” y en caso de éxito se redirigirá al Dashboard, y en caso de fallo se redirigirá a la misma página mostrando un mensaje de error.

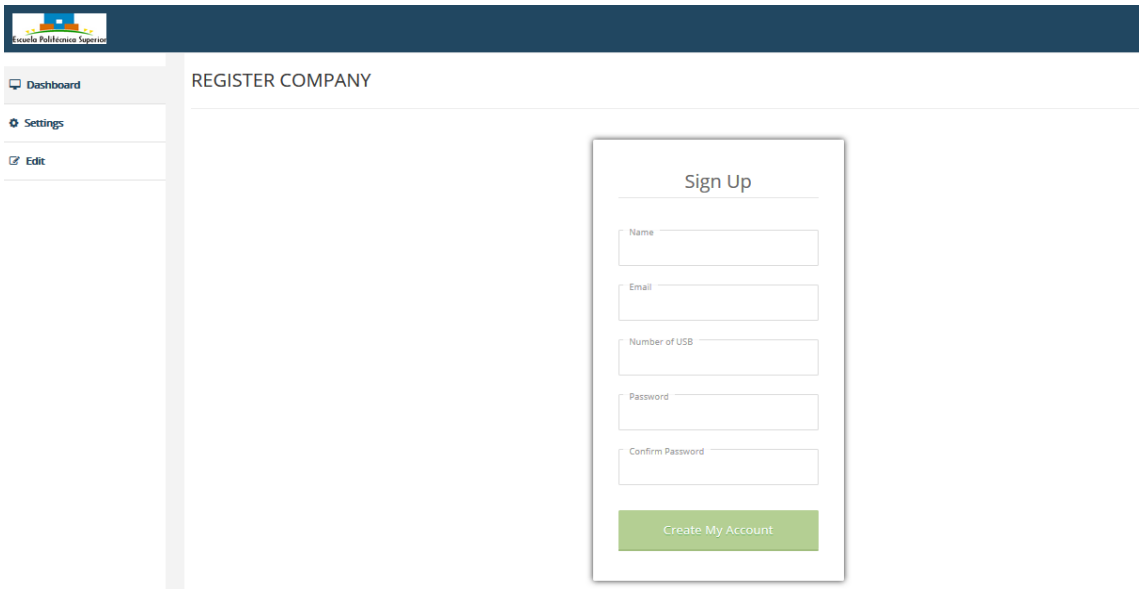


Figura 19: Formulario para dar de alta una nueva campaña

Usuario

Si ingresamos en el login con el rol de usuario, éste nos redirigirá al correspondiente Dashboard como podemos observar en la *Figura 20*.

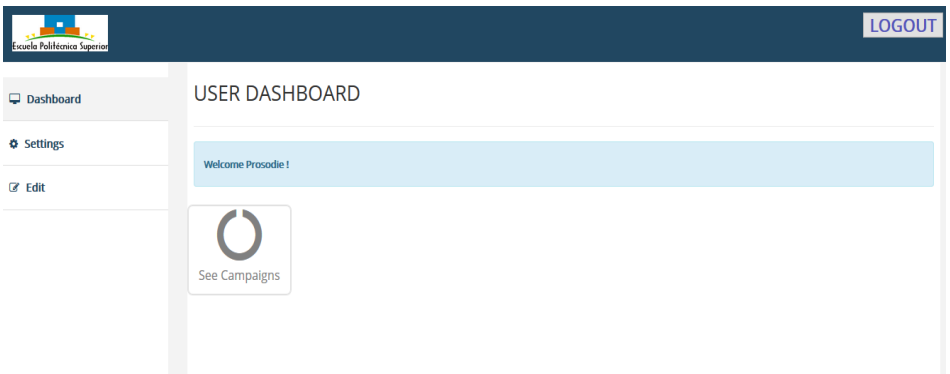


Figura 20: Dashboard correspondiente al rol usuario

En el cual podemos observar que se dispone de la acción See Campaigns, en donde se podrán observar los resultados que se obtienen a tiempo real en la campaña.

Si ingresamos en See Campaigns, podremos observar el siguiente cuadro de mando.

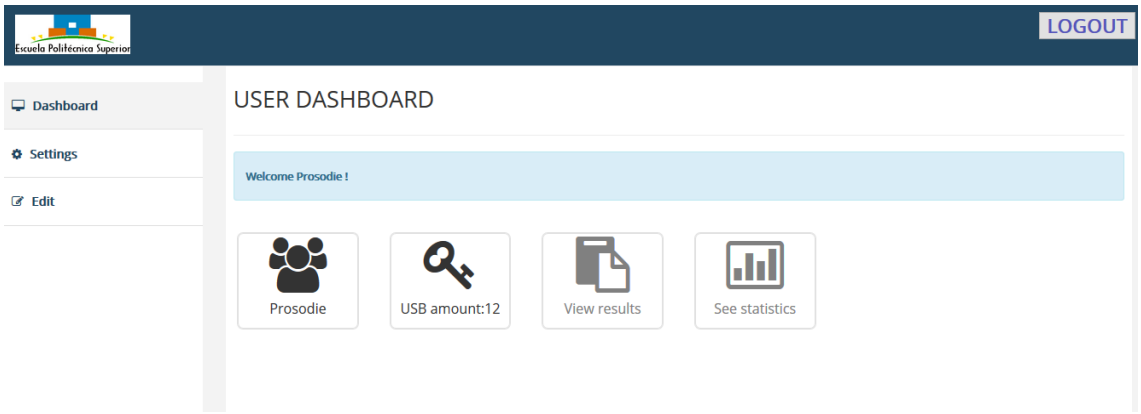


Figura 21: Información general sobre la campaña de un usuario

De modo que se tiene visibilidad del nombre de la empresa y el número de USB´s que se han configurado para la campaña, así como de las posibles opciones de View Results y See Statistics.

Si queremos ver los resultados que se están obteniendo en la campaña, entramos en View Results y nos aparecerá una pantalla como la siguiente (*Figura 22*), en la que se irán reflejando los resultados a medida que se vayan pulsando los archivos maliciosos.

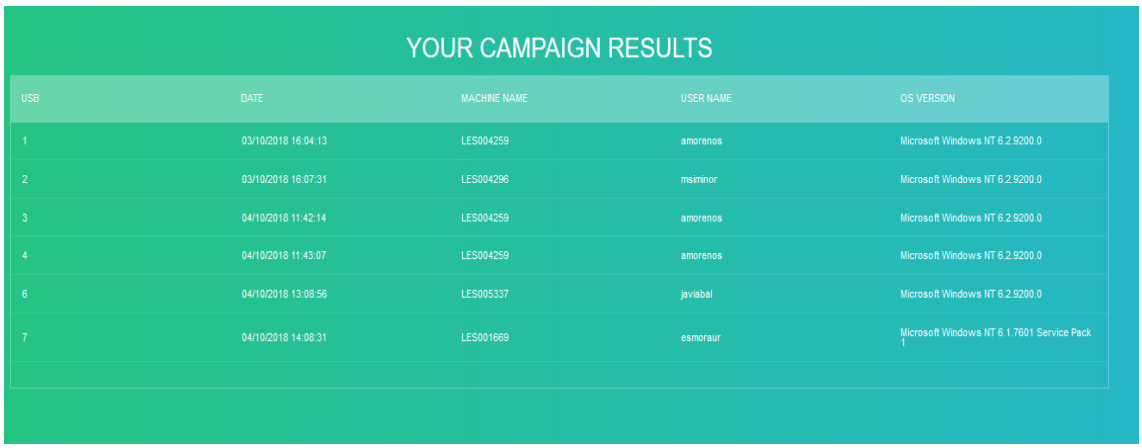
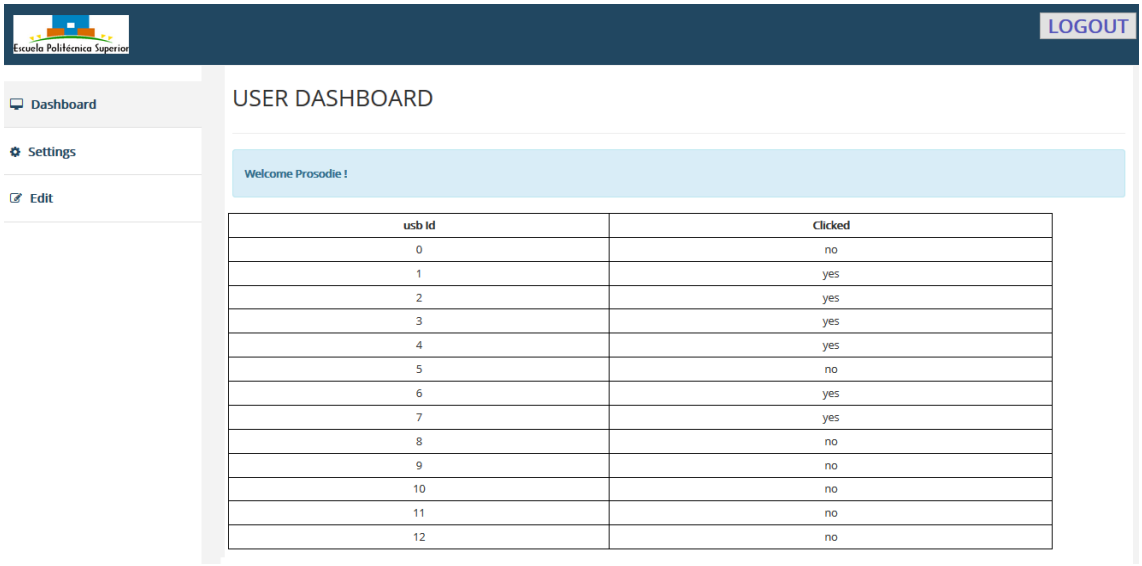


Figura 22: Resultados detallados obtenidos para cada campaña

Si queremos ver en concreto que USB´s han sido los que han “picado” y ver que estadística estamos obteniendo en la campaña, iremos a la opción de See Statistics, en la cual observaremos una pantalla como la siguiente.



usb Id	Clicked
0	no
1	yes
2	yes
3	yes
4	yes
5	no
6	yes
7	yes
8	no
9	no
10	no
11	no
12	no

Figura 23: Resultados estadísticos de cada campaña

En ella podemos ver el número de USB´s configurados para la campaña, y si han pinchado el ejecutable o no. También podremos ver el porcentaje de los USB´s que han sido pulsados frente a los que no.

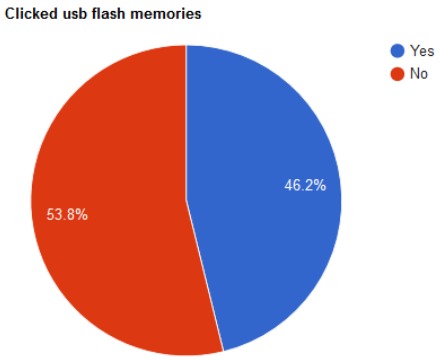


Figura 24: Porcentaje gráfico de USB´s pulsados vs no pulsados

7. PLAN DE PRUEBAS

7.1. Pruebas de verificación

Las pruebas de verificación son aquellas que se realizan para comprobar que todo el proyecto se está construyendo de manera correcta y que los requisitos marcados se satisfacen. En el proyecto hemos realizado varios tipos de pruebas.

Pruebas de caja blanca

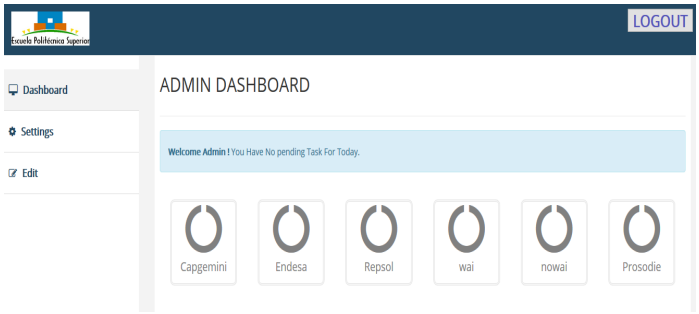
Las pruebas de caja blanca se han realizado dentro de cada operación. Como la mayoría de las operaciones son únicamente una consulta a base de datos, se han realizado directamente las pruebas de caja negra, que explicaremos a continuación. Pero para las operaciones más complejas se ha comprobado el correcto funcionamiento de cada línea de código, que cada bucle se ejecuta el número de veces deseado y que cada consulta a la base de datos nos dé el resultado esperado.

Pruebas de caja negra

Las pruebas de caja negra consisten en hacer pruebas sobres las funciones, en este caso sobre las operaciones, sin conocer su código interior, únicamente introduciendo los parámetros y comprobando la salida.

Para ello, se han creado varias campañas ficticias, en las que se ha ido comprobando que al pulsar los archivos ejecutables los datos se sustraen de forma correcta y podemos tener visibilidad sobre ellos.

Concretamente se han creado las siguientes campañas que aparecen a continuación. Para ello, desde el rol de administrador, registrábamos cada compañía y se cotejaba en la base de datos que se habían creado correctamente.

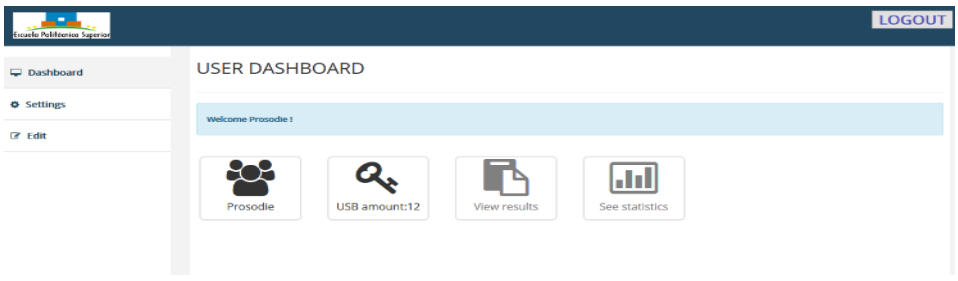


compName	role	id_comp	usbCounter
Capgemini	admin	1	0
Endesa	user	5	17
Repsol	user	7	4
wai	user	9	0
nowai	user	13	0
Prosodie	user	16	12

Figura 25: Total de campañas creadas

Una vez registradas, configuramos las diferentes memorias USB con sus respectivos identificadores para cada campaña creada, y comprobamos que se actualizaban correctamente en la base de datos y que podíamos observarlo en la plataforma.

Por ejemplo, podemos centrarnos en la campaña de prueba creada con el nombre de “Prosodie”, en la cual se han configurado 12 memorias USB como podemos ver en la *Figura 26*.

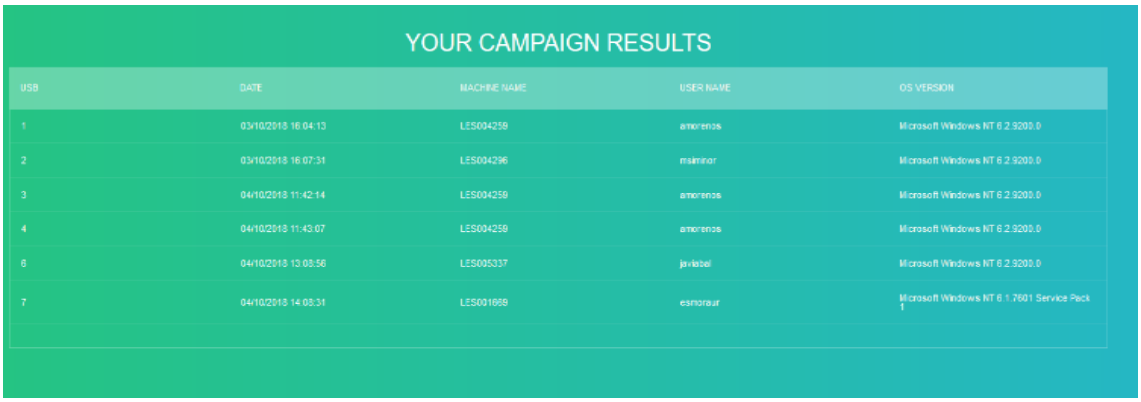


userName	pass	compName	role	id_comp	usbCounter
test@prosodie.com	93fa3e4624676f2e9aa143911118b4547087e9b6e0b6076f2e1027d7a2da2b0a	Prosodie	user	16	12

1 row in set (0.01 sec)

Figura 26: Numero de USB´s configurados para la campaña

De los cuales han pulsado las memorias USB con identificadores 1,2,3,4,6 y 7.




```
mysql> select * from campaigninfo where id_comp=16;
+--------+---------------------+-------------+----------+-----------------------------------------+---------+
| id_usb | Hour                | machineName | userName | osVersion                               | id_comp |
+--------+---------------------+-------------+----------+-----------------------------------------+---------+
| 1      | 03/10/2018 16:04:13 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 16      |
| 2      | 03/10/2018 16:07:31 | LES004296   | msiminor | Microsoft Windows NT 6.2.9200.0         | 16      |
| 3      | 04/10/2018 11:42:14 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 16      |
| 4      | 04/10/2018 11:43:07 | LES004259   | amorenos | Microsoft Windows NT 6.2.9200.0         | 16      |
| 6      | 04/10/2018 13:08:56 | LES005337   | javiabal | Microsoft Windows NT 6.2.9200.0         | 16      |
| 7      | 04/10/2018 14:08:31 | LES001669   | esmoraur | Microsoft Windows NT 6.1.7601 Service Pack 1 | 16      |
+--------+---------------------+-------------+----------+-----------------------------------------+---------+
6 rows in set (0.00 sec)
```

Figura 27: Identificadores de USB´s pulsados en la campaña

Se ha utilizado el mismo procedimiento para el resto de las campañas creadas, y los datos recabados proceden de las pruebas realizadas en los ordenadores corporativos de la empresa en la que trabajo, previo consentimiento de los usuarios.

8. CONCLUSIONES Y TRABAJOS FUTUROS

8.1. Conclusiones

El objetivo de este proyecto era el de proporcionar una herramienta para facilitar, principalmente a las empresas, la evaluación de la concienciación sobre ataques que son muy frecuentes actualmente y pueden tener un impacto severo.

Cabe recalcar que este proyecto tiene el propósito de servir como concienciación y por lo tanto los datos que se sustraen son simplemente para poder identificar a las personas que han mordido el anzuelo y hacerles entender que en estos casos se trata de un estudio pero que los ataques reales siguen el mismo método.

Se tiene como resultado una plataforma que tiene por un lado la programación de un ejecutable que extrae datos sobre las máquinas que pinchan sobre el ejecutable, y por otro lado hemos conseguido crear una herramienta que muestra visualmente los resultados y estadísticas de todos los ejecutables que han sido pulsados en una campaña. Además, también se han establecido los roles de usuario y de administrador, que permiten de una manera cómoda tanto la creación de nuevas campañas como el alta de nuevos usuarios en la herramienta.

8.2. Trabajo futuro

Una posible línea para seguir en el proyecto sería la de perfeccionar un nivel más la plataforma y habilitarle funcionalidades que actualmente se han dejado como estáticas, ya que no era el objetivo del proyecto, como podrían ser la de configurar y editar visualmente los parámetros a extraer en cada campaña, o bien las notificaciones sobre los resultados de una campaña mediante el correo electrónico.

Otra forma de continuar con el trabajo realizado, podría ser la incorporación de nuevos módulos a la herramienta que permitan ver resultados de otro tipo de ataques de ingeniería social como podría ser, por ejemplo, el phishing mediante correos electrónicos.

En definitiva, se trata de una herramienta muy abierta y que permite todo tipo de escalabilidad y perfeccionamiento y que por lo tanto se podrá completar tanto como se desee. A continuación, se facilita un enlace al repositorio público en el cual están depositados todos los ficheros que componen la plataforma y desde donde se podría continuar para trabajos futuros:

https://github.com/alexmore8/Plataforma-USB-dropping

GLOSARIO

3DES *Triple DES*. Algoritmo que hace triple cifrado del DES.

CCTV *Circuito cerrado de televisión*. Es una tecnología de video vigilancia diseñada para supervisar una diversidad de ambientes y actividades.

CSS *Hojas de estilo en cascada*. Es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.

DES *Estándar de cifrado de datos*. Se trata de un sistema de cifrado simétrico por bloques de 64 bits, de los que 8 bits (un byte) se utilizan como control de paridad.

Footprinting *Reconocimiento*. Es donde el atacante obtiene, reúne y organiza toda la información posible sobre su objetivo o su víctima

Firmware *Soporte lógico inalterable*. Es un programa informático que establece la lógica de más bajo nivel que controla los circuitos electrónicos de un dispositivo de cualquier tipo.

Gusano Se trata de un malware que tiene la propiedad de duplicarse a sí mismo.

Hacking Es la búsqueda permanente de conocimientos en todo lo relacionado con sistemas informáticos, sus mecanismos de seguridad y las vulnerabilidades de los mismos.

Hash Función computable mediante un algoritmo, que tiene como entrada un conjunto de elementos, que suelen ser cadenas, y los convierte en un rango de salida finito, normalmente cadenas de longitud fija.

HID *Dispositivo de interfaz humana*. Es un tipo de hardware para la comunicación rápida entre humano-computadora-humano.

HTML *Lenguaje de Marcas de Hipertexto*. Hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia del software que conecta con la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código para la definición de contenido de una página web, como texto, imágenes, videos, juegos, entre otros.

HTTP *Protocolo de Transferencia de Hipertexto*. El protocolo de comunicación que permite las transferencias de información en la World Wide Web. Es un protocolo sin estado, usando las cookies, que es información que un servidor puede almacenar en el sistema cliente.

Interfaz Conexión funcional entre dos sistemas, programas, dispositivos o componentes de cualquier tipo, que proporciona una comunicación de distintos niveles permitiendo el intercambio de información.

Malware *Software malicioso*. Es un tipo de software que tiene como objetivo infiltrarse o dañar una computadora o sistema de información.

PDF *Formato de documento portátil*. Es un formato de almacenamiento para documentos digitales independiente de plataformas de software o hardware.

PHP *Preprocesador de hipertexto*. Es un lenguaje de programación de propósito general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico.

RAM *Memoria de acceso aleatorio*. Memoria de trabajo de computadoras para el sistema operativo, los programas y la mayor parte del software, donde se cargan todas las instrucciones que ejecuta la unidad central de

procesamiento (procesador) y otras unidades del computador.

Sistemas Biométricos Es una tecnología de identificación basada en el reconocimiento de una característica física e intransferible de las personas, como por ejemplo, la huella digital, el reconocimiento del patrón venoso del dedo o el reconocimiento facial.

Sistemas Informáticos Es un sistema que permite almacenar y procesar información; es el conjunto de partes interrelacionadas: hardware, software y personal informático.

Troyano Malware que se presenta al usuario como un programa aparentemente legítimo e inofensivo, pero que, al ejecutarlo, le brinda a un atacante acceso remoto al equipo infectado

Vulnerabilidad Es una debilidad del sistema informático que puede ser utilizada para causar un daño.

XML *Lenguaje de Marcado Extensible*. Lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.

BIBLIOGRAFIA

[1] Kaspersky.*¿Qué es el código malicioso?*, dirección: https://latam.kaspersky.com/resource-center/definitions/malicious-code

[2] Sandoval Castellanos, E. J. *Ingeniería social: corrompiendo la mente humana*, dirección: https://revista.seguridad.unam.mx/numero-10/ingenier%C3%AD-social-corrompiendo-la-mente-humana

[3] Ramiro, R. *Anatomía del ataque de ingeniería social y cómo prevenirlo*, dirección: https://ciberseguridad.blog/anatomia-del-ataque-de-ingenieria-social-y-como-prevenirlo/

[4] Piscitelli, E. *Ingeniería social cuales son los tipos de ataque*, dirección:

http://www.redusers.com/noticias/ingenieria-social-cuales-son-los-tipos-de-ataque/

[5] Informática, N. D*. Diversas Metodologías y tipos de ataque de Ingeniería Social*, dirección: http://noticiasseguridad.com/importantes/diversas-metodologias-y-tipos-de-ataques-de-ingenieria-social/

[6] Enterco*. La Ingeniería Social: El ataque Informático más peligroso*, dirección:

http://www.enter.co/guias/lleva-tu-negocio-a-internet/ingenieria-social/

[7] ABC. *¿Qué es el phishing?*, dirección:

https://www.abc.es/tecnologia/consultorio/abci-phishing-201805170332_noticia.html

[8] Kaspersky. *Ingeniería social, hackeando a personas*, dirección: https://www.kaspersky.es/blog/ingenieria-social-hackeando-a-personas/2066/

[9] Fernández, P. *¿ Cómo puedes ser víctima de un ataque de ingeniería social?*, dirección: https://www.europapress.es/portaltic/ciberseguridad/noticia-puedes-ser-victima-ataque-ingenieria-social-20180209105217.html

[10] Candia, W. M. *Historia y Funcionamiento de la Memoria flash USB*, dirección: http://www.unihorizonte.edu.co/revistas/semilleros/vol/Vol_2_Nro_1_2017/1/Historia_y_Funcionamiento_de_la_Memoria_flash_USB_Willian_Michel_Velez_Candia.pdf

[11] Motos, V. *29 tipos de ataque USB distintos*, dirección: https://www.hackplayers.com/2018/03/29-tipos-de-ataque-usb-distintos.html

[12] Click, D. *¿Qué es el firmware? y mejor aún ¿ para qué sirve?*, dirección: https://dobleclick.eu/que-es-el-firmware-y-para-que-sirve/

[13] Denoizer. *¿Qué es el HID?*, dirección: http://denoizer.com/-que-es-el-hid-.html

[14] Juliá, S. *Los riesgos del uso de la memoria USB en la empresa,* dirección: http://www.gadae.com/blog/uso-riesgos-memoria-usb/

[15] RedTeam. *USB Drop Attacks: The Danger of "Lost And Found" Thumb Drives*, dirección: https://www.redteamsecure.com/usb-drop-attacks-the-danger-of-lost-and-found-thumb-drives/

[16] Arntz, P. *The RTLO method*, dirección: https://blog.malwarebytes.com/cybercrime/2014/01/the-rtlo-method/

[17] Storage, V. a. *RTLO (right to left override) technique for file extension spoofing U+202e*, dirección: https://virtualizationandstorage.wordpress.com/2016/11/17/rtlo-right-to-left-override-technique-for-file-extension-spoofing/

[18] AWS*. Configurar un servidor web Apache y ofrecer archivos de Amazon EFS*, dirección: https://docs.aws.amazon.com/es_es/efs/latest/ug/wt2-apache-web-server.html

[19] INCIBE. *Prevenir los ataques de ingeniería social en las empresas de ocio*, dirección: https://www.incibe.es/protege-tu-empresa/blog/prevenir-los-ataques-ingenieria-social-las-empresas-ocio

APÉNDICE A

JERARQUIA DE DIRECTORIOS

Para ello, se ha diseñado la siguiente jerarquía de directorios:

TFG: Es el directorio principal, que contiene todos los ficheros necesarios.

- **Crypto**: Directorio con todos los ficheros correspondientes al archivo ejecutable.
    - **Bin**: Directorio donde se crea el fichero ejecutable y reside el archivo de configuración conf.xml.
    - **Principal**: Directorio donde se encuentran todos los ficheros C# (.cs).
- **Servidor**: Directorio donde se encuentran todos los ficheros que se encuentran en el servidor.
    - **Base de datos**: Directorio que aloja los scripts para la creación de las tablas en la base de datos.
    - **Aplicación**: Subdirectorio que contiene los ficheros correspondientes a la aplicación y que se divide de la siguiente forma.
        - **Assets**: Directorio que contiene las imágenes y fuentes empleadas en la aplicación.
        - **Css**: Directorio que contiene los ficheros Css que dan apariencia a la aplicación.
        - **Js**: Directorio que contiene los ficheros JavaScript que dan animación a la aplicación.
        - **Scss**: Directorio que contiene el archivo scss utilizado para crear hojas de estilo con un lenguaje que se compila a Css.
        - **Principal**: Directorio que contiene los archivos php y html que son los que realizan el funcionamiento de la aplicación

APÉNDICE B

COMANDOS PARA LA INSTALACIÓN DEL SERVIDOR

En este apartado voy a detallar los pasos que he seguido para crear y configurar el servidor en AWS.

Antes de nada, me gustaría recalcar que se han utilizado todas las funcionalidades gratuitas y en ningún caso se ha comprado ningún tipo de servicio.

Para crear la instancia:

1. Nos registramos en https://aws.amazon.com/ y elegimos la opción Create an AWS Account.
2. Luego ingresamos en el entorno y seleccionamos la opción Launch Instance, en la cual vamos a configurar nuestro servidor.
3. En el primer paso nos dice que tenemos que seleccionar una Imagen y nos proporciona una lista con muchas de ellas, en este caso se ha utilizado un Ubuntu 16.04.
4. En el siguiente paso vamos a configurar algunos detalles de la instancia y vamos a comenzar por la red. Para ello vamos a Network y elegimos la entrada para nuestra VPC predeterminada. Cabe mencionar que, al ser un servicio gratuito, se utiliza el protocolo DHCP y cada vez que se reinicia o se apaga el servidor la IP de nuestro servidor cambia, siempre dentro del rango.
5. En este siguiente paso vamos a ponerle un nombre a nuestra instancia en Tag Instance, en mi caso la he denominado Ubuntu.
6. El siguiente paso es uno de los más importantes y se trata de la configuración de los grupos de seguridad. En este paso tendremos que habilitar el acceso remoto por SSH en el puerto 22, y también debemos habilitar los protocolos HTTP, HTTPS y MySQL de tal forma que el firewall no nos bloquee el tráfico mediante servicios que utilicen esos protocolos.
7. Por último, antes de lanzar nuestra instancia, como estamos utilizando un servidor Ubuntu, es necesario generar un par de claves para realizar

una conexión segura con el servidor y evitar la entrada de desconocidos. Para ellos vamos a la sección de Network & Security y vamos al apartado donde pone Key Pairs. Una vez dentro, seleccionamos create key pair y le damos un nombre al archivo. pem.

Nuestro navegador descargará el archivo de clave privada y es fundamental tenerlo a mano, ya que en caso de pérdida no podremos volver a acceder al servidor.

8. Para finalizar vamos a conectarnos por SSH al servidor. Antes de nada, hay que darle permisos al fichero de clave privada mediante el siguiente comando "*chmod 400 nombredelarchivo.pem*". Una vez tengamos los permisos el comando para ingresar en el servidor sería el siguiente,

"*ssh –i nombredelarchivo.pem nombredelainstancia@ip*". A continuación, muestro un ejemplo del acceso al servidor utilizado para este proyecto.

```
Last login: Fri Aug 31 11:12:41 on console
MacBook-Pro-de-abc:~ Alejandro$ cd Desktop/
MacBook-Pro-de-abc:Desktop Alejandro$ ssh -i serverkey.pem  ubuntu@18.218.211.151
The authenticity of host '18.218.211.151 (18.218.211.151)' can't be established.
ECDSA key fingerprint is SHA256:KMKefTChpcAyZE2hjURX7qNUcCWNvep2dH4JLFBsAFs.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '18.218.211.151' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1062-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

26 packages can be updated.
0 updates are security updates.


Last login: Sun Aug 12 08:52:12 2018 from 88.0.104.244
ubuntu@ip-172-31-38-136:~$
```

Figura 28: Comando para el acceso remoto al servidor

Finalmente, una vez accedí al servidor utilice los comandos aprendidos durante la carrera para crear los diferentes directorios mencionados en el Apéndice A, y utilice el comando scp para mover los ficheros a las correspondientes carpetas del servidor. Un ejemplo sería el siguiente:

"*Scp -i serverkey.pem info.php(fichero) ubuntu@18.222.193.63:/var/www/html (Directorio)*".

APÉNDICE C

PLANIFICACIÓN DEL PROYECTO

A continuación, se detalla la planificación llevada a cabo para el seguimiento del proyecto:

- **Junio 2018**: Planteamiento del problema definiendo su alcance.
- **Julio 2018**: Análisis de requisitos. Se han definido los requisitos del proyecto para así tener una visión global de la funcionalidad que va a tener el proyecto, así como un comienzo del diseño de los componentes del sistema que dan respuesta a las funcionalidades descritas en la etapa de diseño.
- **Agosto 2018**: Selección de los lenguajes de programación adecuados para la realización del proyecto, después de un previo estudio de sus ventajas en el ámbito del proyecto actual. Codificación. Se ha llevado en este período la implementación de los módulos que integran el proyecto, esto es, tanto los módulos que residen en el servidor como el archivo ejecutable.
- **Septiembre 2018**: Pruebas. Período de realización de pruebas unitarias para cada módulo, así como las pruebas de verificación para concretar el correcto funcionamiento del proyecto.
- **Octubre 2018**: Finalización de las pruebas y realización de la memoria correspondiente al proyecto.